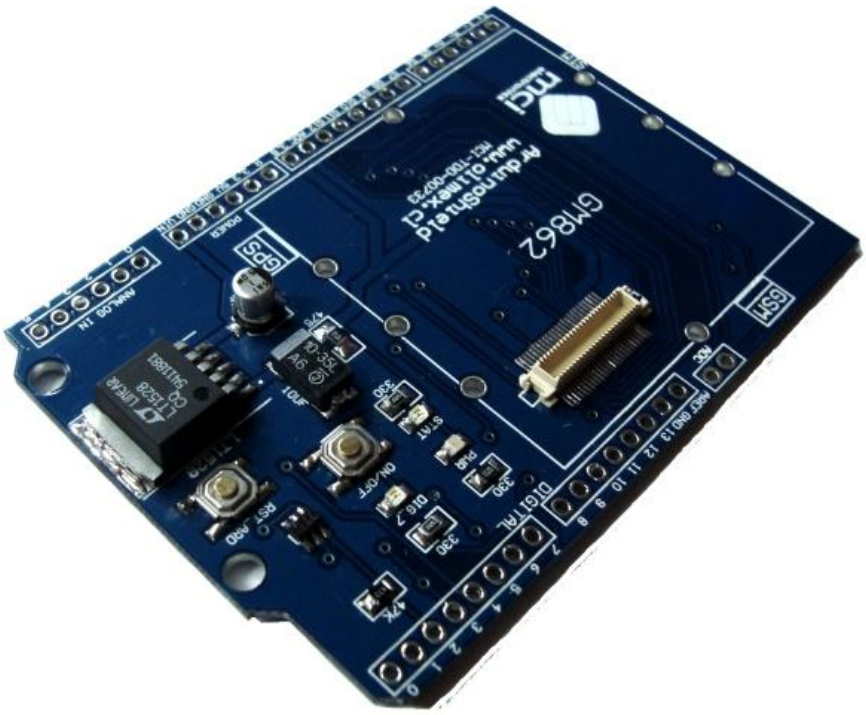
Manual de Usuario

GM862 Arduino Shield

Rev. 1.3

MCI-MA-0063



INGENIERÍA MCI LTDA.

www.olimex.cl
Luis Thayer Ojeda 0115. Of. 402 ▪ Santiago, Chile ▪ Tel. +56 2 3339579 ▪ info@olimex.cl

MCI Ltda.

Luis Thayer Ojeda 0115. Oficina 402
Santiago, Chile

www.olimex.cl

Tel: +56 2 3339579
Fax: +56 2 3350589

® MCI Ltda. 2011

Atención: Cambios y modificaciones hechas en el dispositivo, no autorizados expresamente por MCI, anularán su garantía.

Código Manual: MCI – MA - 0063

INTRODUCCIÓN 4

FUNCIONALIDADES 4

PARTES DEL DISPOSITIVO 5

GUÍA RÁPIDA DE INSTALACIÓN Y PROGRAMACIÓN DE GM862 ARDUINO SHIELD 6

CÓDIGO EJEMPLO GM862 ARDUINO SHIELD 8

MAPA DE PUERTOS 9

HISTORIA DEL DOCUMENTO 13

INTRODUCCIÓN

El objetivo de este documento es la descripción de las funcionalidades del GM862 Arduino Shield.

Este documento abarca todas las funcionalidades que permite el shield, también se sugieren procedimiento de configuraciones básicas del modem GM862.

El GM862 Arduino Shield permite a la tarjeta Arduino hacer llamadas telefónicas, enviar mensajes de texto SMS, enviar datos vía GPRS, accionar pines I/O del módulo y mucho más. Con esto se fusiona la gran cantidad de funcionalidades y flexibilidad de los módems GM862 con la facilidad y sencillez de programación de Arduino. Esto posibilita desarrollar proyectos con Arduino en las siguientes áreas:

- Telemetría y telecontrol
- Sistemas de seguridad inteligentes
- Sistemas de venta automática inteligentes
- Sistemas de gestión de automóviles flota

FUNCIONALIDADES

El GM862 Arduino Shield principalmente entrega la alimentación adecuada al módem Telit GM862 y adapta los pines haciéndolo compatible con Arduino.

Algunas características específicas para éste dispositivo son:

1) Regulación de voltaje a de 5-9 [VDC] a 3.8 [VDC]
2) Consumo máximo de 3 [A]
3) Led indicador de alimentación
4) Led indicador de Status del modem GM862
5) Led configurable por el usuario
6) Botón de encendido/apagado de modem GM862
7) Botón de reset placa Arduino
8) Conexión a Puerto ADC modem GM862
9) Dimensiones: 76 x 54 [mm]

PARTES DEL DISPOSITIVO

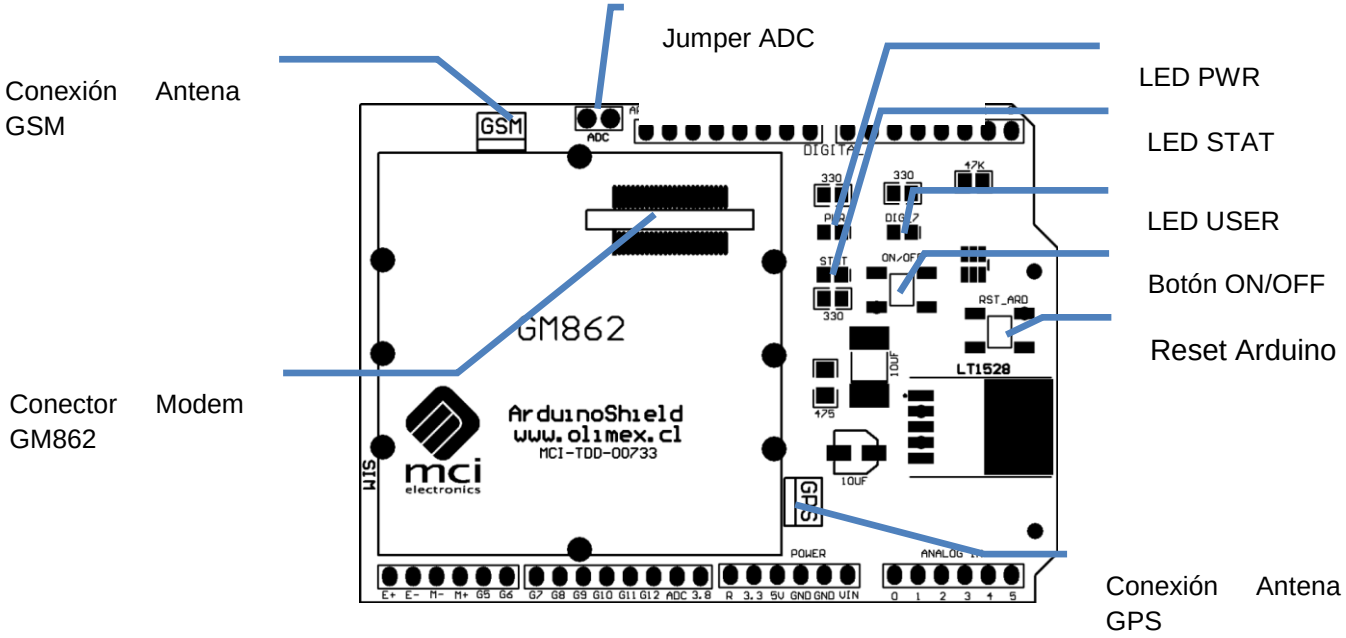


Fig.1 Placa GM862 Arduino Shield

- **Conector Modem GM862**: Conector para modem GM862-GPS
- **LED STAT**: Indicador de funcionamiento de modem GM862
- **LED PWR**: Indicador de alimentación de modem GM862
- **LED USER**: Indicador configurable por el usuario
- **Conector antena GSM**: Conector para antena GSM externa
- **Conector antena GPS:** Conector para antena GPS externa
- **Botón ON/OFF:** Botón de encendido/apagado del modem
- **Botón Reset Arduino**: Resetea placa base Arduino
- **Jumper ADC:** Jumper para setear valor de conversión del ADC del modem GM862 a 0 V.

GUÍA RÁPIDA DE INSTALACIÓN Y PROGRAMACIÓN DE GM862 ARDUINO SHIELD

Para la instalación y programación del GM862 Arduino Shield se debe instalar un puerto serial virtual y conectar algunos pines del Shield, de manera de poder tener comunicación con el modem y con el PC a la vez, permitiendo que el procedimiento de depuración del software sea más fluido que utilizando solamente el puerto serial por hardware. Además, se debe hacer uso de la librería GM862, que contiene funciones elaboradas para poner en marcha la comunicación entre el modulo GM862, arduino, y la red.

1. Descargar programa de ejemplo aquí, y descomprimirlo.
2. Copiar la carpeta newSoftSerial , y la capeta GM862 en la carpeta de instalación del programa arduino, específicamente en libraries.
3. Instalar el Shield. Ver fig. 2 para referencia.
    A. Conectar el modem GM862
    B. Alinear shield de manera de que los pines de conexión de los costados del shield correspondan sean correspondientes con los pines de la placa Arduino, y ensamblar.
4. Insertar SIM a modem GM862
5. Conectar alimentación 5-9VDC a placa Arduino.
6. Conectar puerto USB de Arduino a PC.
7. En caso de que no se hayan instalados los drivers FTDI, instalar drivers USB (Descargar aquí), para tarjetas arduino más recientes, la instalación de los drivers es automática.
8. Como convención, los parámetros que deben ser modificados se podrán identificar por estar escritos con fuente de color rojo, en los Skechers presentados en este manual.

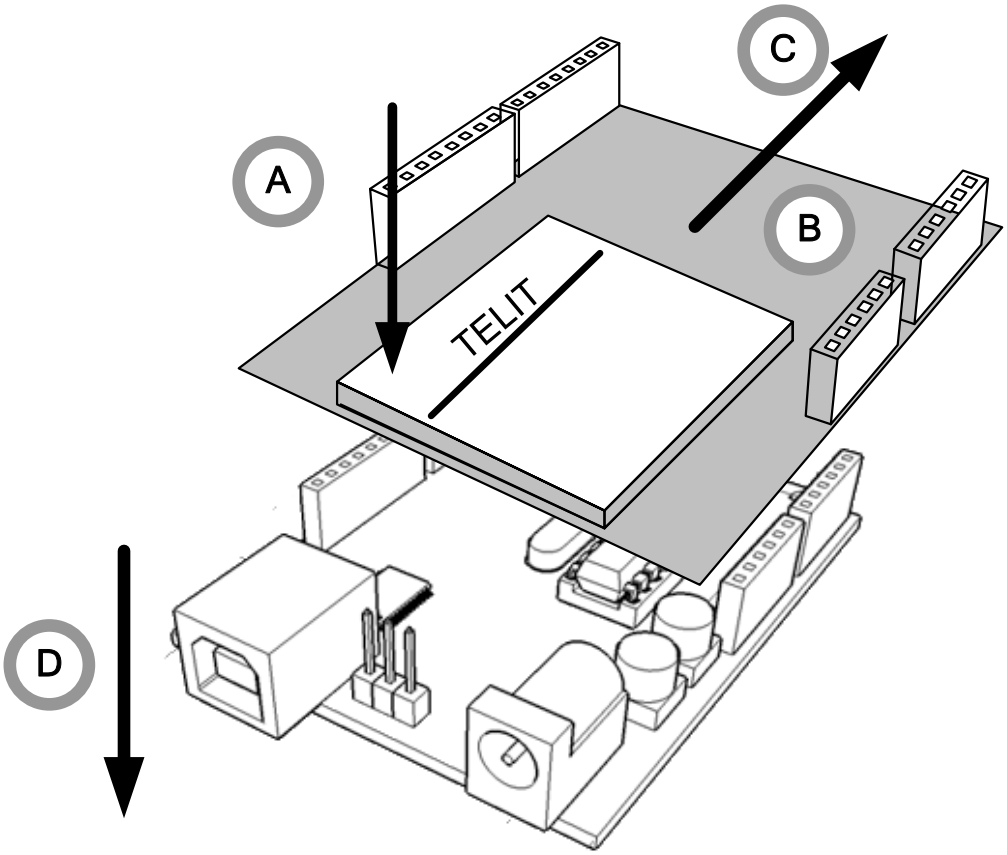


Fig.2 Ensamblaje Placa GM862 Arduino Shield

CÓDIGO EJEMPLO GM862_SMS ARDUINO SHIELD

- Éste código envía un mensaje SMS a un número definido por el programador, con un texto pre-definido. Además realiza toda la inicialización y configuración del módem.
- Los archivos fuente pueden se pueden descargar desde la página del autor o desde www.olimex.cl.
- ¡Es necesario haber realizado los 12 primeros pasos del manual!
- Para cargar el código directamente desde arduino abrir el programa arduino, luego: File>examples>MG862>GM862_SMS.
- Ubicar la función "modem.sendSMS("+569numerotelefonico","mensaje")". Reemplazar el número telefónico, y el mensaje de texto.
- Cargar el programa a arduino.
- Si se abre el visor serial configurado a 9600bds, se pueden apreciar los procedimientos realizados por el microcontrolador.

```
/*
File:       GM862_SMS.pde
Author: Moisés Campos
        MCI Electronics
        www.olimex.cl
*/
#include <NewSoftSerial.h>
#include <string.h>
#include "GM862.h"
#define BUF_LENGTH 100
#define txGprs_pin 3
#define rxGprs_pin 2
NewSoftSerial ser(rxGprs_pin,txGprs_pin);
GM862 modem(&ser);

void setup()
{
   Serial.begin(9600);
   modem.switchOn();
   modem.init();
   while (!modem.isRegistered())
      {
          delay(1000);
          modem.checkNetwork();
      }
}

void loop()
{
   modem.sendSMS("+56998300000", "Hello World!");
   digitalWrite(ST_LED_pin,HIGH);
   delay(2000);
   modem.switchOff();
   while(1);


 }
```

CÓDIGO EJEMPLO GM862_LOAD_HTML ARDUINO SHIELD

- Éste código configura una conexión HTTP con un servidor, en este caso un servidor de prueba del fabricante del modem. Luego de conectarse, el modulo pide un archivo html, el servidor lo envía, el modulo lo carga, todo esto se visualiza en una ventana serial configurada a 9600 bds.
- Para cargar el código directamente desde arduino abrir el programa arduino, luego: File>examples>MG862>GM862_LOAD_HTML.
- Ubicar la función modem.settoInternet("APN","user…….). Reemplazar el número APN, username y password por los parámetros otorgados por su proveedor de servicios de internet movil.
- Cargar el programa a arduino.
- Cargar es programa, y abrir la ventana serial configurada a 9600bds para visualizar los datos.

```
/*
File:       GM862_SMS.pde
Author: Moisés Campos
        MCI Electronics
        www.olimex.cl
*/
#include <NewSoftSerial.h>
#include <string.h>
#include "GM862.h"
#define BUF_LENGTH 100
#define txGprs_pin 3
#define rxGprs_pin 2

NewSoftSerial ser(rxGprs_pin,txGprs_pin);
GM862 modem(&ser);

char generalBuffer[500];

void setup()
{
   Serial.begin(9600);
   modem.switchOn();
   modem.init();
   while (!modem.isRegistered())
      {
          delay(1000);
          modem.checkNetwork();
      }
   modem.settoInternet("APN","username","password","www.roundsolutions.com", "80");
}

void loop()
{
  modem.connecttoServer();
  byte i = 0;
  Serial.println("Load html...");
  modem.sendd("GET / HTTP/1.0\r\n"); // search twitter for gm862
  modem.sendd("Connection: keep-alive\r\n");     // write on the socket
  modem.sendd("\r\n");
  Serial.println("receiving ...");
  while (i++ < 50) {
```

```
    modem.receive(generalBuffer);
    if (strlen(generalBuffer) > 0) {
      Serial.print("buf:");
      Serial.println(generalBuffer);
      i--;
    }
  }
  Serial.println("done");
   modem.closedConection();
   digitalWrite(ST_LED_pin,HIGH);
   modem.switchOff();
   while(1);
}
```

CÓDIGO EJEMPLO GM862_GPRS ARDUINO SHIELD

- Éste código permite abrir un socket con un servidor, por otro lado, lee y procesa datos provenientes de un sensor que ingresan por la entrada analógica 0. Posteriormente envía datos al servidor en tiempo real.
- Para cargar el código directamente desde arduino abrir el programa arduino, luego: File>examples>MG862>GM862_GPRS.
- Ubicar la función modem.settoInternet("APN","user…….). Reemplazar el número APN, username y password por los parámetros otorgados por su proveedor de servicios de internet móvil.
- En la misma función anterior, reemplazar el dominio por el del servidor, y la puerta de acceso.
- Configurar el servidor para revivir datos.
- Cargar el programa, y abrir la ventana serial configurada a 9600bds.

```
/*
File:       GM862_SMS.pde
Author: Moisés Campos
        MCI Electronics
        www.olimex.cl
*/ #include <NewSoftSerial.h>
#include <string.h>
#include "GM862.h"
#define BUF_LENGTH 100
#define txGprs_pin 3
#define rxGprs_pin 2

NewSoftSerial ser(rxGprs_pin,txGprs_pin);
GM862 modem(&ser);

char generalBuffer[100];

void setup()
{
   Serial.begin(9600);
   modem.switchOn();
   modem.init();
   while (!modem.isRegistered())
      {
          delay(1000);
          modem.checkNetwork();
      }
   modem.settoInternet("APN", "username", "password", "domain", "port");
}

void loop()
{
  modem.connecttoServer();
  char iteracion =0;
  while(iteracion < 4)
   {
     char tmp;
     Serial.println("");
     float temperature = float(analogRead(0));
     temperature = processinSensor(temperature,-21.244,1.7908,-0.0278,0.0003,0,(100.0/1024));
     strcpy(generalBuffer,"Temperatura Actual = ");
```

```
        char *pTemporal = generalBuffer + strlen(generalBuffer);
        ftoa(pTemporal,temperature,1);
        modem.transmittoServer(generalBuffer);
        delay(5000);
        iteracion++;
        Serial.println(generalBuffer);
      }
      modem.closedConection();
      digitalWrite(ST_LED_pin,HIGH);
      delay(2000);
      modem.switchOff();
      while(1);
    }

    float processinSensor(float sensor, float a, float ax,float axx, float axxx,float axxxx,
float factor) {
        sensor=(sensor*factor);//valor normalizado, y luego transformado en %
        sensor= (axxxx*pow(sensor,4)+axxx*pow(sensor,3)+axx*pow(sensor,2)+ax*sensor+a);
    return sensor;}

    void *ftoa(char *a, double f, int precision){
      float p[] = {0.0,10.0,100.0,1000.0,10000.0,100000.0,1000000.0,10000000.0,100000000.0};
      char *ret = a;
      char mantiza[10];
      long heiltal = long(f);
      ltoa(heiltal, a, 10);
      while (*a != '\0') a++;
      *a++ = '.';
      long decimal = abs(long((f - float(heiltal)) * p[precision]));
      ltoa(decimal, mantiza, 10);
      for(;strlen(mantiza) < precision; precision--)
        *a++ = '0';
      strcpy(a , mantiza);}
```

MAPA DE PUERTOS

Mapa de pines de GM862 Arduino Shield

Pin	Nombre/Función
DIGITAL 0	Serial RX
DIGITAL 1	Serial TX
DIGITAL 2	SoftSerial RX
DIGITAL 3	SoftSerial TX
DIGITAL 4	ON/OFF Modem
RESET	RESET Ard
EAR_HF+	Handset ear output, phase +
EAR_HF-	Handset ear output, phase -
MIC_HF-	Handsfree microphone input; phase -, nominal level 3mVrms
MIC_HF+	Handsfree microphone input; phase +, nominal level 3mVrms
GPIO5	Configurable general purpose I/O pin
GPIO6	GP I/O pin or ALARM output pin
GPIO7	GP I/O pin or Buzzer output pin
GPIO8	Configurable general purpose I/O pin
GPIO9	Configurable general purpose I/O pin
GPIO10	Configurable general purpose I/O pin
GPIO11	Configurable general purpose I/O pin
GPIO12	Configurable general purpose I/O pin
ADC	ADC input
3.8V	3.8V regulated voltage

HISTORIA DEL DOCUMENTO

Revisión	Fecha	Editado por	Descripción/Cambios
1.0	12 de Agosto 2010	E. Martin	Versión inicial del documento
1.1	22 de Septiembre 2010	E. Martin	Se incluye ejemplo para uso de puerto serial por software
1.2	23 de Noviembre 2010	E. Martin	Actualización de cambios en nueva versión de placa
1.3	21 de Julio 2011	M. Campos	Se incluyen nuevos ejemplos y cambios en las librerías